IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent No. 7,249,228

Confirmation No. 3084

Issued: July 24, 2007

Name of Patentee: Agarwal et al.

Patent Title: REDUCING THE NUMBER OF BLOCK MASKS REQUIRED FOR PROGRAMMING MULTIPLE ACCESS CONTROL LIST IN AN ASSOCIATIVE MEMORY

REQUEST FOR CERTIFICATE OF CORRECTION OF PATENT FOR PATENT OFFICE MISTAKE (37 C.F.R. § 1.322)

Attn: Certificate of Correction Branch
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

It is requested that a Certificate of Correction be issued to correct Office mistakes found the above-identified patent. Attached hereto is a Certificate of Correction which indicates the requested correction. For your convenience, also attached are copies of selected pages (a) from the issued patent with errors highlighted, and (b) from the original application and the declaration as filed March 1, 2004, with the correct text/instructions.

It is believed that there is no charge for this request because applicant or applicants were not responsible for such error, as will be apparent upon a comparison of the issued patent with the application as filed or amended. However, the Assistant Commissioner is hereby authorized to charge any fee that may be required to Deposit Account No. 501430.

Respectfully submitted,
The Law Office of Kirk D. Williams

Date: 11-25-2009
Nov 25, 2009

By _____

Kirk D. Williams, Reg. No. 42,229
One of the Attorneys for Applicants
CUSTOMER NUMBER 26327
The Law Office of Kirk D. Williams
P.O. Box 39425, Denver, CO 80239-0425
303-282-0151 (telephone), 303-778-0748 (facsimile)

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 7,249,228
APPLICATION NO. : 10/791,632
DATED : July 24, 2007
INVENTOR(S) : Agarwal et al.

It is certified that an error appears or errors appear in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Face of the Patent, replace "CONTROL LIST" with – CONTROL LISTS –

Col. 1, Title, "CONTROL LIST" with – CONTROL LISTS –

Col 4, line 21, replace "FIGS. 6A-C" with – FIGS. 6A-F –

Col. 11, line 24, replace "0<=j<=MAX" with – 0<j<=MAX –

Col. 11, line 42, replace "xij1" with – x_{ij} 1 –

Col. 11, line 48, replace "=x_{m-j+1}1" with – x_{m-j+1} –

Col. 11, line 53, replace "+ where" with – + 1 where –

MAILING ADDRESS OF SENDER:
Kirk D. Williams, Reg. No. 42,229
Customer No. 26327
The Law Office of Kirk D. Williams
P.O. Box 39425, Denver, CO 80239

US007249228B1

(12) **United States Patent**
Agarwal et al.

(10) **Patent No.: US 7,249,228 B1**
(45) **Date of Patent: Jul. 24, 2007**

(54) **REDUCING THE NUMBER OF BLOCK MASKS REQUIRED FOR PROGRAMMING MULTIPLE ACCESS CONTROL LIST IN AN ASSOCIATIVE MEMORY**

(75) Inventors: **Amit Agarwal**, Union City, CA (US); **Venkateshwar Rao Pullela**, San Jose, CA (US); **Qizhong Chen**, Palo Alto, CA (US)

(73) Assignee: **Cisco Technology, Inc.**, San Jose, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 522 days.

(21) Appl. No.: **10/791,632**

(22) Filed: **Mar. 1, 2004**

(51) **Int. Cl.**
G06F 12/00 (2006.01)

(52) **U.S. Cl.** **711/154**; 711/100; 711/108; 718/100; 718/102; 718/103; 718/104; 370/395.7; 370/395.72; 370/411; 370/412; 370/413; 370/414; 370/415; 370/416; 719/320

(58) **Field of Classification Search** 711/108, 711/154
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,187,790	A *	2/1993	East et al.	719/316
6,295,576	B1	9/2001	Ogura et al.	
6,374,326	B1	4/2002	Kansal et al.	
6,377,577	B1	4/2002	Bechtolsheim et al.	
6,389,506	B1	5/2002	Ross et al.	
6,467,019	B1	10/2002	Washburn	
6,526,474	B1	2/2003	Ross	
6,535,951	B1	3/2003	Ross	
6,606,681	B1	8/2003	Uzun	
6,643,260	B1	11/2003	Kloth et al.	
6,651,096	B1	11/2003	Gai et al.	
6,658,002	B1	12/2003	Ross et al.	
6,658,458	B1	12/2003	Gai et al	
6,687,144	B2	2/2004	Batson et al	
6,715,029	B1	3/2004	Trainin et al.	
6,717,946	B1	4/2004	Hariguchi et al.	
6,725,326	B1	4/2004	Patra et al.	
6,738,862	B1	5/2004	Ross et al.	
6,775,737	B1	8/2004	Warkhede et al.	
6,839,256	B1 *	1/2005	Preebsting et al.	365/49
6,862,281	B1	3/2005	Chandraseksran	
6,871,262	B1	3/2005	Oren et al.	
6,871,265	B1	3/2005	Oren et al.	
6,910,097	B1 *	6/2005	Srinivasan et al.	711/108
6,961,808	B1	11/2005	Oren et al.	
6,970,971	B1	11/2005	Warkhede et al.	
7,002,965	B1 *	2/2006	Cheriton	370/395.32
2003/0231631	A1	12/2003	Pullela	
2004/0030802	A1	2/2004	Eatherton et al.	
2004/0030803	A1	2/2004	Eatherton et al.	
2004/0100950	A1	5/2004	Basu et al.	
2004/0170171	A1	9/2004	Kanekar et al.	
2004/0170172	A1	9/2004	Pullela et al.	
2004/0172346	A1	9/2004	Kanekar et al.	
2005/0010612	A1	1/2005	Enderwick et al.	
2005/0114602	A1	5/2005	Ngai et al.	
2005/0157712	A1	7/2005	Rangaranjan et al.	

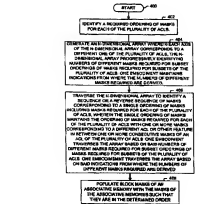
* cited by examiner

Primary Examiner—Sanjiv Shah
Assistant Examiner—Zhuo H. Li
(74) *Attorney, Agent, or Firm*—The Law Office of Kirk D. Williams

(57) **ABSTRACT**

Mechanisms for reducing the number of block masks required for programming multiple access control lists in an associative memory are disclosed. A combined ordering of masks corresponding to multiple access control lists (ACLs) is typically identified, with the multiple ACLs including n ACLs. An n-dimensional array is generated, wherein each axis of the n-dimensional array corresponds to masks in their requisite order of a different one of the multiple ACLs. The n-dimensional array progressively identifies numbers of different masks required for subset orderings of masks required for subsets of the multiple ACLs. The n-dimensional array is traversed to identify a sequence of masks corresponding to a single ordering of masks including masks required for each of the multiple ACLs.

30 Claims, 16 Drawing Sheets



REDUCING THE NUMBER OF BLOCK MASKS REQUIRED FOR PROGRAMMING MULTIPLE ACCESS CONTROL LISTS IN AN ASSOCIATIVE MEMORY

TECHNICAL FIELD

One embodiment of the invention relates to communications and computer systems, especially networked routers, packet switching systems, and other devices using associative memories (e.g., content-addressable memories); and more particularly, one embodiment relates to reducing the number of block masks required for programming multiple access control lists in an associative memory.

BACKGROUND

The communications industry is rapidly changing to adjust to emerging technologies and ever increasing customer demand. This customer demand for new applications and increased performance of existing applications is driving communications network and system providers to employ networks and systems having greater speed and capacity (e.g., greater bandwidth). In trying to achieve these goals, a common approach taken by many communications providers is to use packet switching technology. Increasingly, public and private communications networks are being built and expanded using various packet technologies, such as Internet Protocol (IP).

A network device, such as a switch or router, typically receives, processes, and forwards or discards a packet based on one or more criteria, including the type of protocol used by the packet, addresses of the packet (e.g., source, destination, group), and type or quality of service requested. Additionally, one or more security operations are typically performed on each packet. But before these operations can be performed, a packet classification operation must typically be performed on the packet.

Packet classification as required for, inter alia, access control lists (ACLs) and forwarding decisions, is a demanding part of switch and router design. The packet classification of a received packet is increasingly becoming more difficult due to ever increasing packet rates and number of packet classifications. For example, ACLs typically require matching packets on a subset of fields of the packet header or flow label, with the semantics of a sequential search through the ACL rules.

Access control and quality of service features are typically implemented based on programming contained in one or more ACLs. To implement features in hardware, these multiple ACL lists are typically combined into one list, which can be used for programming and associative memory. Various techniques are known for combining these items, such as Binary Decision Diagram (BDD) and Order Dependent Merge (ODM). For example, if there are two ACLs A (having entries A1 and A2) and B (having entries B1 and B2), then ODM combines these original lists to produce one of two cross-product equivalent ordered lists, each with four entries: A1B1, A1B2, A2B1, and A2B2; or A1B1, A2B1, A1B2, and A2B2. These four entries can then be programmed into an associative memory and an indication of a corresponding action to be taken placed in an adjunct memory. Lookup operations can then be performed on the associative and adjunct memories to identify a corresponding action to use for a particular packet being processed. There are also variants of ODM and BDD which may filter out the entries which are unnecessary as their values will never allow them to be matched. Merged entries which are order independent can be sorted based on common masks, and programmed into the block masks of an associative memory (which typically does not significantly reduce the number of block masks required), or can be programmed in any order in an associative memory where each entry has its own mask field. Nonconsecutive merged entries which remain order dependent must maintain their ordering when programmed into an associative memory, and thus cannot be rearranged to reduce or eliminate redundant masks when entries are masked using block masks. Also, one or more of these techniques may produce an increased number of entries and/or block masks required for programming the resultant entries into an associative memory.

An example of an associative memory using block masks is described in Ross et al., "Block Mask Ternary CAM", U.S. Pat. No. 6,389,506, issued May 12, 2002, which is hereby incorporated by reference. In a nutshell, a block mask is a mask that is applied to each entry of a block of entries. Such an associative memory typically has numerous blocks and corresponding block masks. FIG. **1A** shows one such prior art associative memory **100**, having multiple blocks **110**, **120**, and **130**, each with corresponding block masks **111**, **121**, and **131** for blocks of associative memory entries **112**, **122**, and **132**.

FIG. **1B** illustrates a prior art approach for combining masks of two ACLs **150** and **152**, having masks as shown with their corresponding required ordering. The result of a first approach for combining these lists is shown in ordering **155**, in which entries of ACL-**2** **152** are concatenated at the end of entries of ACL-**1** **150** to produce an ordering that requires m masks, where m is the sum of the number of masks required for each of ACLs **150** and **152**. The results **156** and **157** of a second approach is similar, but allows the mask at the end of a list to be used by both ACLs **150** and **152** if the last required mask of one ACL is the same mask as first required by the other ACL, then the number of masks required is m minus a small number of overlapping masks. However, this does not significantly reduce the overall number of masks required, which can be a problem as the number of different masks in the required order is directly correlated to the number of ACL entries which can be stored in a block mask associative memory. Thus, an efficient way of allocating these masks is desired.

SUMMARY

Disclosed are, inter alia, methods, apparatus, data structures, computer-readable medium, mechanisms, and means for reducing the number of block masks required for programming multiple access control lists in an associative memory.

One embodiment identifies a combined ordering of masks corresponding to multiple access control lists (ACLs), the multiple ACLs including n ACLs. A required ordering of masks for each of the n ACLs is identified. An n-dimensional array is generated, wherein each axis of the n-dimensional array corresponds to masks in their requisite order of a different one of the multiple ACLs. The n-dimensional array progressively identifies numbers of different masks required for subset orderings of masks required for subsets of the multiple ACLs. The n-dimensional array is traversed to identify a sequence (e.g., the order or reverse order) of masks corresponding to a single ordering of masks including masks required for each of the multiple ACLs. The single ordering of masks maintains the ordering of masks required for each of the multiple ACLs with one or more masks

corresponding to a different ACL or other feature in between one or more consecutive masks of an ACL of the multiple ACLs.

In one embodiment, a last position identified by a last column and last row of the array identifies the number of different masks required for the single ordering of masks. In one embodiment, the matrix is traversed based on said numbers of different masks required for subset orderings of masks required for subsets of the multiple ACLs. One embodiment maintains indications from where said numbers of different masks required for subset orderings of masks required for subsets of the plurality of ACLs are generated, and the n-dimensional array is traversed based on said indications from where said numbers of different masks required for subset orderings of masks required for subsets of the plurality of ACLs are generated One embodiment populates multiple block masks of an associative memory with said masks required for the multiple ACLs such that the single ordering of masks is produced in the associative memory. Rather than combining all n ACLs at a time when n is greater than two, one embodiment successively combines two ACLs together, then combines that result with a next ACL, and so on.

One embodiment identifies a combined ordering of masks corresponding to a first ACL and a second ACL. A first ordering of masks required for the first ACL is identified. A second ordering of masks required the second ACL is identified. A matrix of the first and second orderings of masks is generated, with the matrix progressively identifying numbers of different masks required for subset orderings of masks required for subsets of the first and second ACLs. The matrix is traversed to identify a sequence (e.g., the order or reverse order) of masks corresponding to a single ordering of masks including masks required for the first ACL and the second ACL. The single ordering of masks maintains the first ordering and second orderings of masks with one or more masks corresponding to a different ACL or other feature in between one or more consecutive masks of the first and second ACLs.

In one embodiment, a last position identified by a last column and last row of the matrix identifies the number of different masks required for the single ordering of masks. In one embodiment, the matrix is traversed based on said numbers of different masks required for subset orderings of masks required for subsets of the first and second ACLs. One embodiment maintains indications from where said numbers of different masks required for subset orderings of masks required for subsets of the first and second ACLs are generated, and the matrix is traversed based on said indications from where said numbers of different masks required for subset orderings of masks required for subsets of the first and second ACLs are generated. One embodiment populates multiple block masks of an associative memory with said masks required for the first and second ACLs such that the single ordering of masks is produced in the associative memory.

BRIEF DESCRIPTION OF THE DRAWINGS

The appended claims set forth the features of the invention with particularity. The invention, together with its advantages, may be best understood from the following detailed description taken in conjunction with the accompanying drawings of which:

FIGS. **1A-B** are block diagrams illustrating a prior art associative memory with block masks and prior art approaches for combining masks from two access control lists;

FIGS. **2A-2D** illustrate the generation as performed in one embodiment of an array/matrix progressively identifying the number of different masks required for subset orderings of masks required for subsets of multiple ACLs;

FIGS. **2E-F** illustrate the traversal as performed in one embodiment of an array/matrix to identify a mask ordering;

FIGS. **3A-C** illustrate an array/matrix generated and traversed in one embodiment to identify a mask ordering;

FIG. **4A** is a flow diagram illustrating a process used in one embodiment for generating and traversing an array/matrix;

FIG. **4B** is a flow diagram illustrating a process used in one embodiment for generating and traversing an array/matrix;

FIG. **5** illustrates pseudo-code used in one embodiment for generating and traversing an array/matrix; and

FIGS. **6A-C** are block diagrams of various exemplary systems including one or more embodiments for reducing the number of block masks required for programming multiple access control lists in an associative memory and/or for performing lookup operations on the programmed associative memories.

(3) Should be "FIGS 6A-F"

DETAILED DESCRIPTION

Disclosed are, inter alia, methods, apparatus, data structures, computer-readable medium, mechanisms, and means for reducing the number of block masks required for programming multiple access control lists in an associative memory.

Embodiments described herein include various elements and limitations, with no one element or limitation contemplated as being a critical element or limitation. Each of the claims individually recites an aspect of the invention in its entirety. Moreover, some embodiments described may include, but are not limited to, inter alia, systems, networks, integrated circuit chips, embedded processors, ASICs, methods, and computer-readable medium containing instructions. One or multiple systems, devices, components, etc. may comprise one or more embodiments, which may include some elements or limitations of a claim being performed by the same or different systems, devices, components, etc. The embodiments described hereinafter embody various aspects and configurations within the scope and spirit of the invention, with the figures illustrating exemplary and non-limiting configurations.

As used herein, the term "packet" refers to packets of all types or any other units of information or data, including, but not limited to, fixed length cells and variable length packets, each of which may or may not be divisible into smaller packets or cells. The term "packet" as used herein also refers to both the packet itself or a packet indication, such as, but not limited to all or part of a packet or packet header, a data structure value, pointer or index, or any other part or direct or indirect identification of a packet or information associated therewith. For example, often times a router operates on one or more fields of a packet, especially the header, so the body of the packet is often stored in a separate memory while the packet header is manipulated, and based on the results of the processing of the packet (i.e., the packet header in this example), the entire packet is forwarded or dropped, etc. Additionally, these packets may contain one or more types of information, including, but not limited to, voice,

consecutive masks of the first and second ACLs. In one embodiment, a last position identified by a last column and last row of the matrix identifies the number of different masks required for the single ordering of masks. In one embodiment, the matrix is traversed based on said numbers of different masks required for subset orderings of masks required for subsets of the first and second ACLs. In one embodiment, the matrix is traversed based on said indications of where the numbers of different masks are derived.

One embodiment populates multiple block masks of an associative memory with said masks required for the first and second ACLs such that the single ordering of masks is produced in the associative memory.

Finally, in process block **448**, the multiple block masks of an associative memory are populated with the masks required for the multiple ACLs. Processing is complete as indicated by process block **450**.

Another way of viewing the identification of the ordering of masks is to define a cost function Cost(X, n), where X is the ACL, MAX is the maximum number of value entries with each mask entry. Thus,

(4 Should be .0< j ≤ MAX-

$$Cost(X,n)=0 \text{ if } n=0 \quad Cost(X,n-j)+1 \text{ where } x_n=x_{n-1}=\ldots=x_{n-j+1} \text{ and } 0\leqq j\leqq MAX \text{ if } n>0$$

Problem statement: Given the ACLs

$$X_1=<x_{11},x_{12},x_{13},\ldots x_{1n_1}>$$

$$X_2=<x_{21},x_{22},x_{23},\ldots,x_{2n_2}>$$

$$\ldots$$

$$X_m=<x_{m1},x_{m2},x_{m3},\ldots,x_{mn_m}>$$

Compute Z, which is an ordering of the aces in X_1, X_2, . . . X_m where

$$Z=<z_1,z_2,\ldots,z_{(n1+n2+\ldots+n_m)}>$$

such that

(5) Should be. ∀xij|-

$\forall xij1\leqq i\leqq m, 1\leqq j\leqq n, \exists z_k$ where $1\leqq k\leqq(n_1+n_2+\ldots+n_m)$ and

if $x_{ij}=z_a$ and $x_{ik}=z_b$ and j<k then a<b

and Cost(Z, $n_1+n_2+n_3+\ldots+n_m$) is minimal.

Thus, for two ACLs, m=2 in the above problem statement and it reduces as follows.

(6) Should be =x_{m-j}+1

$$fs(X,m)=0 \text{ if } m=0 \; j \text{ where } x_m=x_{m-1}=\ldots=x_{m-j+1} \text{ and } 1\leqq j\leqq MAX$$

(7) should be +1 where

$$V(X,Y:m,n)= \qquad \text{Equation 1}$$

$$\text{Min}(V(X,Y:m-a,n), V(X,Y:m,n-b)) + \text{where } x_m \neq y_n \quad \underset{\forall i,j:i+j\leq MAX,i\leq a,j\leq b}{\text{Min}}(V(X,Y:m-i,n-j)) + \text{where } x_m=y_n \text{ where } a=f_s(X,m), b=f_s(Y,n)$$

For two ACLs, X_1 and X_2, m=2 and the solution to the above recurrence relation $V(X_1, X_2: x_{1n_1}, x_{2n_2})$ gives the optimal number of masks required. FIG. **5** illustrates pseudo-code **500** for generating and traversing a matrix to identify the ordering of the masks to use. The process illustrated in pseudo-code 400 is a formalization of that previously described herein, so this discussion will not be repeated.

The time requirement for the algorithm [Min_Masks and Find_Optimized_ACL] is O(mn) where m is the number of aces in ACL_a and n is the number of aces in ACL_b. The space requirement is O(mn).

This algorithm can be easily extended to more than two ACLs as illustrated in the pseudo-code below. Using the above approach the time requirement is $O(m_1m_2m_3\ldots m_n)$ where ACL_a ha m_1 aces, ACL_b has m_2 aces . . . and ACL_n has m_n aces. The space requirement is $O(m_1m_2\ldots m_n)$.

```
Result⇐NULL
for (i=1; i<no of ACLs; i++)
  Result⇐{Min_Masks(Result, Acl_i); Find_Optimized_Acl(Result, ACL_i)};
```

With the above the time requirement is the order of

$$O\left(\sum_{j=1}^{n-1} mj \cdot \sum_{i=j+1}^{n} mi\right)$$

which is $O(m^3)$ when $m_1=m_2=\ldots=m_n$ and the space requirement is $O(nm^2)$.

FIGS. **6**A-F are block diagrams of various exemplary systems including one or more embodiments for reducing the number of block masks required for programming multiple access control lists in an associative memory and/or for performing lookup operations on the programmed associative memories. First, FIG. **6**A illustrates one embodiment of a system, which may be part of a router or other communications or computer system, for determining a reduced number of block masks, for programming corresponding entries and block masks in one or more associative memories, and for performing lookup operations to produce results which can be used in the processing of packets. In one embodiment, control logic **610** determines the required ordering of block masks for multiple ACLs and, via signals **611**, programs and updates associative memory or memories **615**. In one embodiment, control logic **610** also programs memory **620** via signals **623**. In one embodiment, control logic **610** includes custom circuitry, such as, but not limited to discrete circuitry, ASICs, memory devices, processors, etc.

In one embodiment, packets **601** are received by packet processor **605**. In addition to other operations (e.g., packet routing, security, etc.), packet processor **605** typically generates one or more items, including, but not limited to one or more packet flow identifiers based on one or more fields of one or more of the received packets **601** and possibly from information stored in data structures or acquired from other sources. Packet processor **605** typically generates a lookup value **603** which is provided to control logic **610** for providing control and data information to associative memory or memories **615**, which perform lookup operations and generate one or more results **617**. In one embodiment, a result **617** is used is by memory **620** to produce a result **625**. Control logic **610** then relays result **607**, based on result **617** and/or result **625**, to packet processor **605**. In response, one or more of the received packets are manipulated and forwarded by packet processor **605** as indicated by packets **609**. Note, results **617**, **625** and **607** may include indications of error conditions.

FIG. **6**B illustrates one embodiment of a system, which may be part of a router or other communications or computer system, for determining a reduced number of block masks,

PATENT

Attorney's Docket No. **15654**

COMBINED DECLARATION AND POWER OF ATTORNEY

As below named inventor, I hereby declare that

This declaration is of the following type: ORIGINAL

My residence, post office address, and citizenship are as stated below next to my name. I believe I am the original, first, and sole inventor *(if only one name is listed below)* or an original, first, and joint inventor *(if plural names are listed below)* of the subject matter which is claimed and for which a patent is sought on the invention entitled:

(1) (2)

TITLE Appears incorrectly 2 places in Issued patent - Face of the Patent and at top of Col. 1

REDUCING THE NUMBER OF BLOCK MASKS REQUIRED FOR PROGRAMMING MULTIPLE ACCESS CONTROL LISTS IN AN ASSOCIATIVE MEMORY

the specification of which is attached hereto.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claim(s), as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, § 1.56.

As a named inventor, I hereby appoint **Kirk D. Williams, Reg. 42,229** to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

I further direct that correspondence concerning this application be directed to:

CUSTOMER NUMBER 26327
Kirk D. Williams, Esq.
The Law Office of Kirk D. Williams
1234 S. OGDEN ST.
Denver, CO 80210
303-282-0151 (telephone)
303-778-0748 (facsimile)

I hereby declare that all statements made herein of my own knowledge are true, that all statements made on information and belief are believed to be true, that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Full name of sole or first inventor: **AMIT AGARWAL**

Inventor's signature Agarwal

Date 02/26/2004 Country of Citizenship: **INDIA**

Residence: **UNION CITY, CA**
Post Office Address: **3236 SANTA PAULA WAY, UNION CITY, CA 94587**

FROM PATENT APPLICATION FILED 03-01-2004

REDUCING THE NUMBER OF BLOCK MASKS REQUIRED FOR PROGRAMMING MULTIPLE ACCESS CONTROL LISTS IN AN ASSOCIATIVE MEMORY

(1) & (2) Appears incorrectly on Patent Title Page and Col. 1

TECHNICAL FIELD

One embodiment of the invention relates to communications and computer systems, especially networked routers, packet switching systems, and other devices using associative memories (e.g., content-addressable memories); and more particularly, one embodiment relates to reducing the number of block masks required for programming multiple access control lists in an associative memory.

BACKGROUND

The communications industry is rapidly changing to adjust to emerging technologies and ever increasing customer demand. This customer demand for new applications and increased performance of existing applications is driving communications network and system providers to employ networks and systems having greater speed and capacity (e.g., greater bandwidth). In trying to achieve these goals, a common approach taken by many communications providers is to use packet switching technology. Increasingly, public and private communications networks are being built and expanded using various packet technologies, such as Internet Protocol (IP).

A network device, such as a switch or router, typically receives, processes, and forwards or discards a packet based on one or more criteria, including the type of protocol used by the packet, addresses of the packet (e.g., source, destination, group), and type or quality of service requested. Additionally, one or more security operations are typically performed on each packet. But before these operations can be performed, a packet classification operation must typically be performed on the packet.

Packet classification as required for, *inter alia,* access control lists (ACLs) and forwarding decisions, is a demanding part of switch and router design. The packet

FROM PATENT APPLICATION FILED 03-01-2004

BRIEF DESCRIPTION OF THE DRAWINGS

The appended claims set forth the features of the invention with particularity. The invention, together with its advantages, may be best understood from the following detailed description taken in conjunction with the accompanying drawings of which:

FIGs. 1A-B are block diagrams illustrating a prior art associative memory with block masks and prior art approaches for combining masks from two access control lists;

FIGs. 2A-2D illustrate the generation as performed in one embodiment of an array/matrix progressively identifying the number of different masks required for subset orderings of masks required for subsets of multiple ACLs;

FIGs. 2E-F illustrate the traversal as performed in one embodiment of an array/matrix to identify a mask ordering;

FIGs. 3A-C illustrate an array/matrix generated and traversed in one embodiment to identify a mask ordering;

FIG. 4A is a flow diagram illustrating a process used in one embodiment for generating and traversing an array/matrix;

FIG. 4B is a flow diagram illustrating a process used in one embodiment for generating and traversing an array/matrix;

FIG. 5 illustrates pseudo-code used in one embodiment for generating and traversing an array/matrix; and

FIGs. 6A-F are block diagrams of various exemplary systems including one or more embodiments for reducing the number of block masks required for programming multiple access control lists in an associative memory and/or for performing lookup operations on the programmed associative memories.

(3) See Col. 4 line 21

PRIOR PATENT APPLICATION FILED 03-01-2004

One embodiment populates multiple block masks of an associative memory with said masks required for the first and second ACLs such that the single ordering of masks is produced in the associative memory.

Finally, in process block 448, the multiple block masks of an associative memory are populated with the masks required for the multiple ACLs. Processing is complete as indicated by process block 450.

Another way of viewing the identification of the ordering of masks is to define a cost function *Cost(X, n)*, where *X* is the ACL, *MAX* is the maximum number of value entries with each mask entry. Thus,

(4) See Col. 11 line 24

$$Cost(X,n) = 0 \text{ if } n = 0$$

$$Cost(X,n-j)+1 \text{ where } x_n = x_{n-1} = \ldots = x_{n-j+1} \text{ and } 0 < j \le MAX \text{ if } n > 0$$

Problem statement: Given the ACLs

$X_1 =< x_{11}, x_{12}, x_{13}, \ldots, x_{1n1} >$

$X_2 =< x_{21}, x_{22}, x_{23}, \ldots, x_{2n2} >$

...

$X_m =< x_{m1}, x_{m2}, x_{m3}, \ldots, x_{mnm} >$

Compute *Z*, which is an ordering of the aces in $X_1, X_2, \ldots, X_m$ where

(5) See Col. 11, line 41

$Z =< z_1, z_2, \ldots, z_{(n1+n2+\ldots+nm)} >$ such that

$\forall x_{ij}\ 1 \le i \le m, 1 \le j \le n_i \ \exists\ z_k$ where $1 \le k \le (n_1+n_2+\ldots+n_m)$ and

if $x_{ij} = z_a$ and $x_{ik} = z_b$ and $j < k$ then $a < b$

and $Cost(Z, n_1+n_2+n_3+\ldots+n_m)$ is minimal.

Thus, for two ACLs, m=2 in the above problem statement and it reduces as follows.

(6) See Col. 11, line 48

$$fs(X,m) = 0 \text{ if } m = 0$$

$$j \text{ where } x_m = x_{m-1} = \ldots (= x_{m-j+1}) \text{ and } 1 \le j \le MAX$$

FROM PATENT APPLICATION FILED 03-01-2004

(7)

$$V(X,Y:m,n) = Min(V(X,Y:m-a,n),V(X,Y:m,n-b)) + 1 \text{ where } x_m \neq y_n$$
$$\underset{\forall i,j,i+j \leq MAX, i \leq a, j \leq b}{Min}(V(X,Y:m-i,n-j)) + 1 \text{ where } x_m = y_n$$
$$\text{where } a = f_1(X,m), b = f_1(Y,n) \quad \text{... Equation 1}$$

See Col. 11, Line 54

For two ACLs, X_1 and X_2, m=2 and the solution to the above recurrence relation $V(X_1, X_2 : x_{1n_1}, x_{2n_2})$ gives the optimal number of masks required. FIG. 5 illustrates pseudo-code 500 for generating and traversing a matrix to identify the ordering of the masks to use. The process illustrated in pseudo-code 400 is a formalization of that previously described herein, so this discussion will not be repeated.

The time requirement for the algorithm [Min_Masks and Find_Optimized_ACL] is O(mn) where m is the number of aces in ACL_a and n is the number of aces in ACL_b. The space requirement is O(mn).

This algorithm can be easily extended to more than two ACLs as illustrated in the pseudo-code below. Using the above approach the time requirement is $O(m_1m_2m_3...m_n)$ where ACL_a ha m_1 aces, ACL_b has m_2 aces … and ACL_n has m_n aces. The space requirement is $O(m_1m_2...m_n)$.

Result ⇐ *NULL*

for (i=1; i< no of ACLs; i++)

Result ⇐ *{Min_Masks(Result, Acl$_i$) ; Find_Optimized_Acl(Result, ACL$_i$)};*

With the above the time requirement is the order of $O\left(\sum_{j=1}^{n-1} m_j \cdot \sum_{i=j+1}^{n} m_i\right)$ which is $O(m^3)$ when $m_1=m_2=...=m_n$ and the space requirement is $O(nm^2)$.

FIGs. 6A-F are block diagrams of various exemplary systems including one or more embodiments for reducing the number of block masks required for programming multiple access control lists in an associative memory and/or for performing lookup operations on the programmed associative memories. First, FIG. 6A illustrates one embodiment of a system, which may be part of a router or other communications or